

Active Flutter Suppression Using Reduced-Order Modeling for Transonic Aeroservoelastic Control Law Development

Josiah M. Waite,^{*} Bret K. Stanford,[†] Walter A. Silva,[†] and Robert E. Bartels[‡]
NASA Langley Research Center, Hampton, VA, 23681

In this paper, several aerodynamic reduced-order models (ROMs) are generated and coupled with structural models to form aeroelastic ROMs. The aerodynamic ROMs generated here include the effects of control surface motion and are appropriate for use in aeroservoelastic applications. Simple observer-based full-state feedback controllers were designed from these aeroelastic ROMs. Additionally, observer gain matrices were designed from and coupled to the aeroelastic ROMs. Each (linear) observer was then used to estimate the dynamics of a (nonlinear) stand-alone computational fluid-structure dynamics simulation. Then, using the estimated states and the full-state feedback controller, control surface commands were fed back into the computational fluid-structure dynamics simulation to successfully achieve active flutter suppression. The process, as well as some results, are presented in this paper.

Nomenclature

Acronyms

AFS = Active flutter suppression
 ASE = Aeroservoelastic
 CFD = Computational fluid dynamics
 CRM = Common research model
 ROM = Reduced-order model

Symbols

A = State matrix
 B = Input matrix
 b = Wingspan
 C = Output matrix
 D = Feedforward matrix
 K = Feedback gain matrix
 L = Observer gain matrix
 m = Generalized mass vector
 q = Dynamic pressure
 u = Input vector
 x = State vector
 y = Wing spanwise coordinates
 η = Generalized displacement vector
 δ = Physical displacement vector
 ω = Natural frequency vector
 ζ = Damping ratio

Subscripts

a = Aerodynamic
 c = Control
 s = Structural

^{*}Research Aerospace Engineer, Aeroelasticity Branch, MS 340, AIAA Member, josiah.m.waite@nasa.gov

[†]Research Aerospace Engineer, Aeroelasticity Branch, MS 340, AIAA Associate Fellow

[‡]Research Aerospace Engineer, Aeroelasticity Branch, MS 340, AIAA Senior Member

I. Introduction

Active flutter suppression (AFS) for a fixed-wing aircraft involves the use of active control surfaces to artificially expand the flutter instability boundary of a flexible wing via closed-loop control. The success of this technique depends heavily on the guarantee that the control system, including the control law and the hardware in the control system, does not fail. If this cannot be guaranteed, the likelihood of experiencing flutter increases, often beyond acceptable risk limits. This has historically limited the use of AFS systems to military and other non-civilian aircraft. Despite the risk, however, AFS has received much attention from several researchers [1]. This is, at least in part, because AFS offers the possibility of reduced threats of flutter-based failures that could allow for lighter aircraft structures, which could lead to significant fuel cost savings.

There have been several studies, both experimental and numerical in nature, that have explored the use of AFS [2–6]. When numerical studies of flutter suppression have been conducted, they have traditionally been performed within the framework of an aeroservoelastic (ASE) model that includes computational models for the structure, the fluid, and some control law. Typically, a linear formulation (such as a doublet lattice method, or DLM) has been used for the fluid solver to keep the computational cost of the analysis down [5, 7–10]. The reduction in computational cost, however, often comes at the price of reduced accuracy in predicting the fluid dynamics, particularly for transonic flows. Since modeling aeroelastic phenomena in the transonic flight regime is of interest to many researchers, this inability to capture transonic aerodynamic nonlinearities limits the problems to which the ASE model can be applied.

Furthermore, as aircraft designs tend toward wings with higher aspect ratio and increasing flexibility, interactions between structural dynamics and aerodynamics become more complex and the ability to generate accurate, nonlinear aerodynamic predictions becomes essential to aircraft design [11]. Although researchers have successfully integrated higher-order CFD solvers into computational aeroelastic models [12, 13], it is still considered to be a challenge because of the high computational cost [14], particularly for unsteady simulations. Because of this, there are few dynamic aeroelastic analyses done using higher-order CFD solvers, and most of those are done using simple geometries. There are even fewer ASE models that integrate CFD solvers [15–17].

Reduced-order modeling is an alternative to directly implementing CFD solvers into aeroelastic analysis frameworks [18]. These reduced-order models (ROMs) are typically (but not always [19–21]) linearized systems derived from CFD data and so contain information regarding the nonlinear behavior of the fluid, which makes them particularly well-suited as analysis tools for transonic flows [2, 19, 21–23]. If the ROM has been properly formulated, parameters within the ROM can be changed and results for new configurations or conditions can be generated in much less time than would be required by additional CFD simulations [20, 24].

In this work, several aerodynamic ROMs, which were valid for transonic Mach numbers, were generated and coupled with structural models to form aeroelastic ROMs. Simple observer-based full-state feedback controllers were designed from these aeroelastic ROMs. Additionally, observer gain matrices were designed from and coupled to the aeroelastic ROMs. Each (linear) observer was then used to estimate the dynamics of a (nonlinear) stand-alone computational fluid-structure dynamics simulation. Then, using the estimated states and the full-state feedback controller, control surface commands were fed back into the computational fluid-structure dynamics simulation. Thus, AFS was demonstrated on a jet transport configuration at transonic cruise speeds.

Because the aerodynamic ROM was trained at a transonic Mach number, it captured nonlinearities in the flow that could not be captured by linear aerodynamics. Additionally, it ran in a fraction of the time that a traditional CFD simulation would take. Furthermore, it has been shown that, for a given set of controller gains, aeroservoelastic models that employ an aerodynamic ROM that captures flow nonlinearities better than a linear aerodynamic model can lead to more effective AFS systems [2]. These facts make the approach outlined here desirable for AFS at transonic speeds. The purposes of this paper are to (1) outline the procedure for developing a ROM-based AFS system suitable for transonic flight speeds and (2) demonstrate and assess the performance of several AFS systems based on different linear aeroelastic models. This assessment will guide future directions of the work.

The procedure can be broken down into three phases: aerodynamic training simulation, ROM generation, and observer and controller design. Details of each of these phases, along with a brief description of the geometry, will be discussed in Sec. II. Section III will compare the performance of several AFS controllers based on different linear aeroelastic models. Finally, several conclusions regarding the work and future directions will be discussed in Sec. IV.

II. Methods

A. Overview

The overall method will be presented here first, followed by discussion of each step. This work was completed in three phases that fed into one another: (1) aerodynamic training simulation, (2) ROM generation, (3) observer and controller design. The first phase, running an aerodynamic training simulation, produced sets of input-output data to be used in the second phase. In the second phase, the data from the training simulation was used to generate one or more aerodynamic ROMs. A structural dynamics model was combined with the aerodynamic ROM(s) to produce an aeroelastic state-space model. The last phase added a controller to the aeroelastic model to produce an aeroservoelastic model. This aeroservoelastic model was then used as an observer to estimate the dynamics of a plant model (which was, in this case, a computational aeroelastic simulation) and compute control surface deflections that would suppress flutter, if present. A block diagram of the final aeroservoelastic system and its implementation as an observer is shown in Fig. 1. In this diagram, and throughout the rest of this paper, η_s and $\ddot{\eta}_s$ are modal displacement and modal acceleration of structural modes of the system as predicted by the plant; δ represents physical vertical accelerations at specific points along the wing; \hat{x} and $\hat{\dot{x}}$ are estimates of the linear aeroelastic states of a stand-alone linear aeroelastic model of the plant as predicted by the observer; η_c represents the modal displacements of control surfaces; $A, B, C,$ and D are the aeroelastic state-space matrices derived in the second phase, mentioned above; L is an observer gain matrix which, when combined with the aeroelastic state-space matrices, constitute the observer; and K is the feedback gain matrix, or controller.

The analyses presented here were conducted on a semispan model of the NASA Common Research Model (CRM; see Fig. 2) as developed in Ref. [25]. This is a generic transport geometry with a cruise Mach number of 0.85, wing span of 58.7 meters, mean aerodynamic chord of 7 meters, aspect ratio of 9, taper ratio of 0.275, and sweep angle of 35°. Results presented in this paper are at a Mach number of 0.85 and angle of attack of 0°. Two trailing-edge control surfaces were attached to the wing as shown in the bottom of Fig. 2. When deflected, the edges of each trailing-edge control surface were blended into the trailing edge of the wing, as shown in the top right of Fig. 2.

Only the wing was considered to be flexible in this work, populated with an aluminum wingbox built from ribs, spars, skins, and stiffeners, described more fully in Ref. [25]. The wingbox was connected to a series of lumped masses meant to emulate fuel and engine inertia. A total of 25 flexible structural modes and two rigid control modes (where the control surfaces were assumed to be irreversible, or rigid with rigid hinges) were generated for this geometry and are used by the plant in its aeroelastic simulations. To keep the computational cost down, fewer modes were used when generating the aeroelastic ROMs. Modes 1, 2, 3, and 8 were used to generate the aeroelastic ROMs presented in this paper as they were the most influential modes in the vehicle’s overall motion, an insight obtained from linear aeroelastic analysis. The two control modes represented a unit rotation in the control surface while holding the rest of the wing fixed.

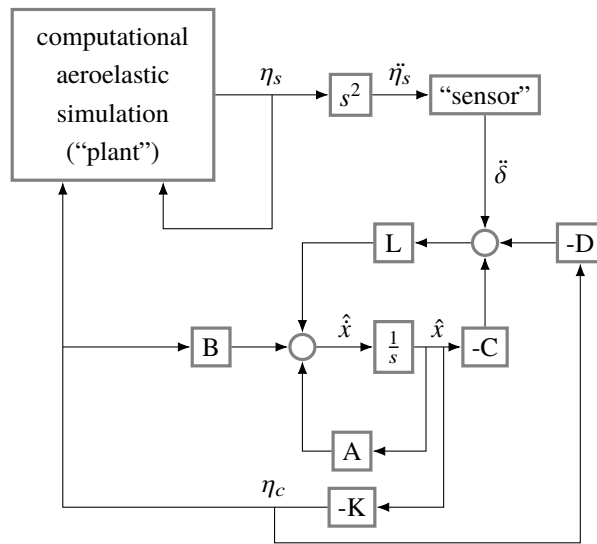


Fig. 1 Block diagram of the final aeroservoelastic system and its implementation as an observer to the plant.

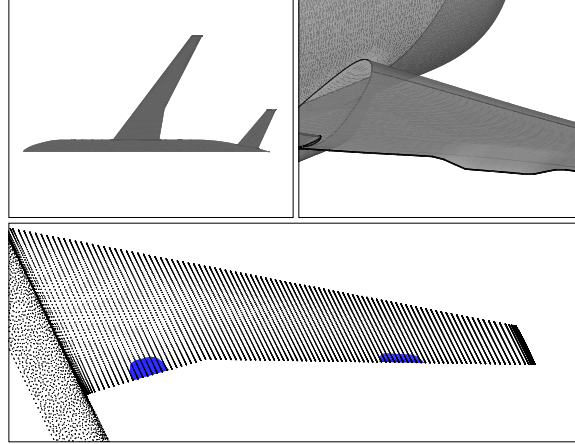


Fig. 2 Top left: Planform view of the NASA CRM semispan model used in the current analysis. Top right: Illustration of one control surface deflection. Note the control surface blending. Bottom: Representation of the two control surfaces used in the current model, with control surface #1 near the root and control surface #2 near the tip.

The control surfaces associated with the two control modes were located at $2y/b = 0.214$ and 0.787 . Visualizations are presented here for modes 1, 2, 3, and 8. Figure 3 shows (qualitatively) the vertical deflection magnitudes for modes 1, 2, 3, and 8, which had natural frequencies of 10.7, 23.6, 26.4, and 85.0 rad/s, respectively.

As stated earlier, one of the purposes of this work was to assess the performance of AFS systems based on several different linear aeroelastic models. This was accomplished using the framework shown in Fig. 1 with the geometry just described. The rest of this section will provide more detailed discussion of each phase.

B. Phase 1: Aerodynamic Training Simulation

A training simulation that provided input-output data to a system identification algorithm was the first step in generating an aerodynamic ROM. The NASA Langley Research Center FUN3D [26] code was used to conduct these training simulations. FUN3D is primarily a computational fluid dynamics solver, but it also has a built-in modal structural dynamics solver that uses a second-order accurate predictor-corrector scheme. This capability to simulate modal aeroelastic dynamics was utilized in the training simulations to prescribe the motion of specific structural and/or control modes and to calculate the resulting aerodynamic forces on the structural modes. For all FUN3D analyses, the unsteady Reynolds-averaged Navier-Stokes equations were solved using a second-order upwind scheme with a Spalart-Allmaras turbulence model. A fairly coarse semi-infinite tetrahedral volume mesh was used here, with 3.1 million nodes, and a $y+$ value nearly 1 throughout the surface. No validation or verification data will be presented here for the computational mesh beyond a mesh convergence study shown in Ref. [27].

As a first step in setting up the training simulation, a converged steady flow solution with a rigid aircraft configuration was obtained. Because it was desired that the final aerodynamic ROM be able to exchange modal displacements and

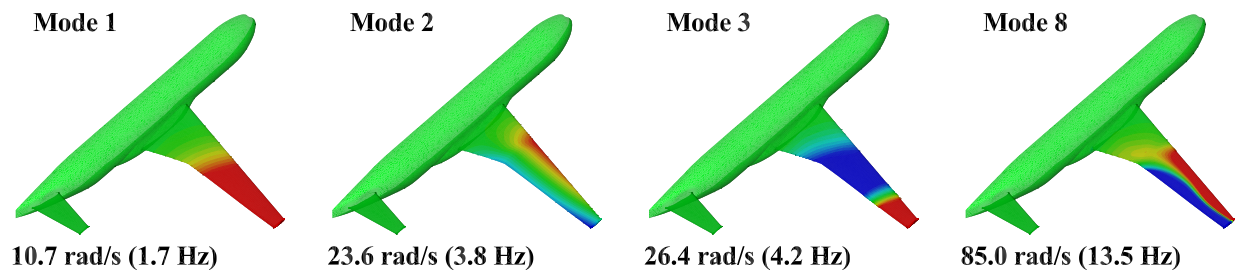


Fig. 3 Vertical deflections and frequencies associated with modes 1, 2, 3, and 8.

aerodynamic forces with a structural dynamics model, it was necessary to set up a training simulation whose inputs were modal displacements and whose outputs included modal aerodynamic forces. To do this, a set of modes was defined that would be used to represent the structural dynamics of the vehicle. For this work, the vehicle’s structural dynamics were represented by 4 structural modes and 2 control surface modes, as mentioned in Section II.A. A static aeroelastic simulation was then restarted from the converged steady solution in which the control modes were not allowed to evolve, but the structural modes were evolved until static aeroelastic convergence was achieved.

The converged static aeroelastic solution was then the starting point for another simulation where, instead of holding control modes fixed at zero and allowing structural modes to respond to the flow, the motion of all the modes (or some subset of the modes) was prescribed and the output of interest was the aerodynamic forces acting on the structural modes. Which modes are excited depends on what the final aerodynamic ROM is designed to capture. For this work, three types of aerodynamic ROMs were generated: one where all 6 modes were excited together, another where only the 4 structural modes were excited, and, lastly, one where only the 2 control modes were excited. Additionally, the type of prescribed motion can vary as long as the inputs are orthogonal. For example, the prescribed motion could be Gaussian pulses or sine waves, as long as all signals are orthogonal to one another. Here, orthogonal Walsh functions [28] were used to prescribe the motion of each mode. An example of Walsh function inputs that might be used for a case where all 6 modes are excited is shown in Fig. 4. The length of the Walsh inputs and the step size used were tailored to capture a desired frequency range. Note that these two parameters, along with the number of modes being excited simultaneously, will often have to be balanced against available resources as they affect the duration of the simulation. Determining the magnitudes of the Walsh functions is less straightforward than determining their lengths, as this quantity is often configuration-dependent. In the cases presented here, the magnitudes of excitations to the structural modes were based on converged static aeroelastic deflections, while magnitudes of excitations to control modes were kept between 1° and 3° .

In order to assess the sensitivity of the ROM to changes in the conditions of the training simulation, two parameters within the training simulation were varied: (1) the dynamic pressure at which the training simulation was executed (2) and the amplitude of the modal excitations. Varying the dynamic pressure led to differences in static aeroelastic deflections, which led to differences in shock strength and location. These variations resulted in four training simulation conditions, presented here as:

- “a” - This was the baseline condition, run at a dynamic pressure of 16.9 kPa. (Note that the flutter point for this configuration was calculated previously to be approximately 30.3 kPa [27].) The training simulation had Walsh input amplitudes of 2° for the control surfaces and 5% of the static aeroelastic point for each of the flexible modes (for example, if the static displacement for mode 1 was 100, then the Walsh amplitude applied mode 1 for the training simulation was 5).

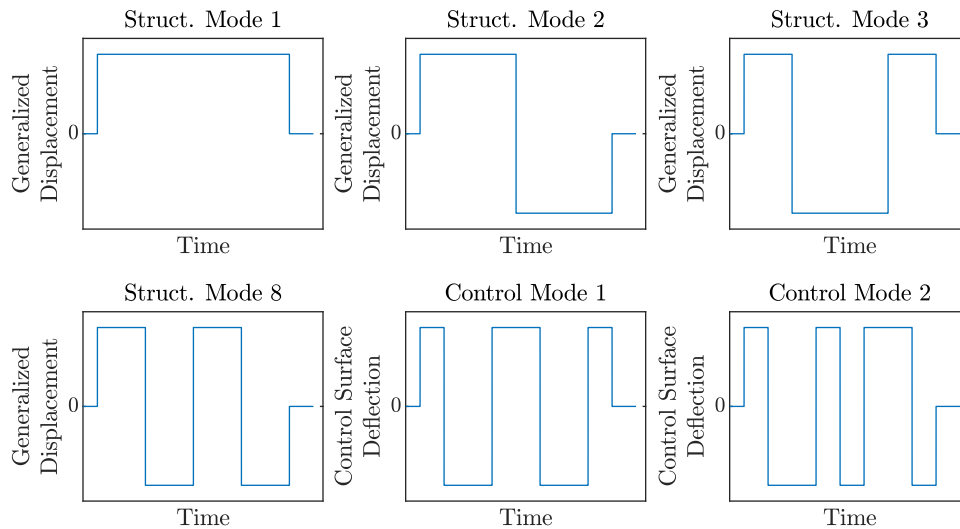


Fig. 4 Excitation profiles for the four structural modes and two control modes used in the Walsh function training simulation.

- “b” - This training simulation was run at an identical dynamic pressure as “a”, but the ratio of control mode excitation amplitude to structural mode excitation amplitude was increased.
- “c” - This training simulation was run at a dynamic pressure 50% higher than “a”, but with the same ratio of control mode excitation amplitude to structural mode excitation amplitude as “a”.
- “d” - This training simulation was run at both higher dynamic pressure and higher ratio of control mode excitation amplitude to structural mode excitation amplitude relative to “a”.

These variations are denoted in the results by appending identifiers to each ROM (for example, ROM-IIa represents the ROM-II model whose training simulation conditions correspond to “a” above, and so on).

For each training simulation, the output was a set of modal force responses of the structural modes due to some inputs, which were prescribed modal displacements in each specified mode.

C. Phase 2: Reduced-Order Model Generation

Once a training simulation was completed, the input-output data, which in this case were modal displacement-modal aerodynamic force data, was passed to AEROM [22]. AEROM is a suite of tools developed at the NASA Langley Research Center particularly for generating aeroelastic reduced-order models. Two tools in particular from AEROM were used in this work: the PULSE algorithm and the eigensystem realization algorithm (ERA). The PULSE algorithm was used to generate individual impulse responses of all inputs on all outputs from the input-output data produced by the training simulation. A brief overview of how this algorithm works is given in Ref. [22]. A discrete-time state-space model of the system was then generated via ERA. A description of ERA is given in Ref. [29]. For the present case, PULSE and ERA were used to generate an aerodynamic state-space ROM whose inputs were modal displacements (of control surfaces, the structure, or a combination of the two) and whose outputs were modal aerodynamic forces on the structure. The final equations were as follows:

$$\begin{aligned}\{\hat{x}_a\} &= [A_a] \{\hat{x}_a\} + [B_a] \{u_a\} \\ \{y_a\} &= [C_a] \{\hat{x}_a\} + [D_a] \{u_a\}\end{aligned}\tag{1}$$

where

$$\{u_a\} = \begin{Bmatrix} \hat{\eta}_s \\ \eta_c \end{Bmatrix}$$

and $\{\hat{x}_a\}$ are the aerodynamic states from the ROM. Note that the overhats are used here to emphasize that these quantities are estimates of quantities from a stand-alone linear aeroelastic model of the plant, and this notation is consistent with Fig. 1.

Once the aerodynamic model was formed, a linear representation of the structural dynamics was formed. Inputs to the structural model include generalized masses, modal frequencies, and modal damping. For the cases presented in this paper, these parameters for the 4 structural modes came from NASTRAN. The final structural dynamics equations took the form:

$$\begin{aligned}\{\hat{x}_s\} &= [A_s] \{\hat{x}_s\} + [B_s] \{u_s\} \\ \{y_s\} &= [C_s] \{\hat{x}_s\}\end{aligned}\tag{2}$$

where

$$\begin{aligned}\{u_s\} &= q \{y_a\} \\ \{\hat{x}_s\} &= \begin{Bmatrix} \hat{\eta}_s \\ \hat{\eta}_s \end{Bmatrix}\end{aligned}$$

$$[A_s] = \begin{bmatrix} 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ -\omega_1^2 & 0 & 0 & 0 & -2\zeta\omega_1 & 0 & 0 & 0 \\ 0 & -\omega_2^2 & 0 & 0 & 0 & -2\zeta\omega_2 & 0 & 0 \\ 0 & 0 & -\omega_3^2 & 0 & 0 & 0 & -2\zeta\omega_3 & 0 \\ 0 & 0 & 0 & -\omega_4^2 & 0 & 0 & 0 & -2\zeta\omega_4 \end{bmatrix}$$

$$[B_s] = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 1/m_1 & 0 & 0 & 0 \\ 0 & 1/m_2 & 0 & 0 \\ 0 & 0 & 1/m_3 & 0 \\ 0 & 0 & 0 & 1/m_4 \end{bmatrix}$$

$$[C_s] = \begin{bmatrix} 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}.$$

Here, ζ , the structural damping present in the system, was held to zero.

Having established aerodynamic and structural models, attention is now turned to coupling them in order to form a unified aeroelastic model. Two different families of aeroelastic ROMs were developed and are described as ROM-II and ROM-III in Ref. [30]. A brief overview of how these two ROM families were assembled will be given here, keeping the same names as in Ref. [30].

The first family of ROMs, designated ROM-II, was assembled as shown in Fig. 5. The aerodynamic ROM was derived from a training simulation where all control and structural modes were excited together. To develop the combined aeroelastic state-space system, it was first necessary to account for the fact that the aerodynamic ROM input

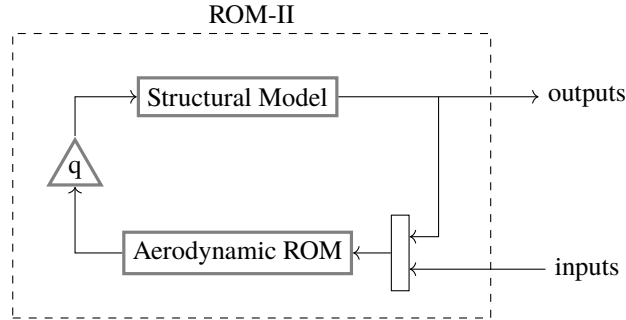


Fig. 5 Block diagram for ROM-II. Inputs for ROM-II were two control mode displacements and outputs were modal displacements of four modes of interest.

vector was of a different length than the structural output vector. This was accounted for by defining

$$\begin{aligned} [H_1] &= \begin{bmatrix} I \\ 0 \end{bmatrix} \\ [H_2] &= \begin{bmatrix} 0 \\ I \end{bmatrix} \end{aligned}$$

and redefining:

$$\{u_a\} = [H_1 C_s] \{\hat{x}_s\} + [H_2] \{\eta_c\}. \quad (3)$$

After substituting Eq. 3 into Eq. 1 and combining with Eq. 2, and after defining the estimated aeroelastic states as

$$\{\hat{x}\} = \begin{Bmatrix} \hat{x}_a \\ \hat{x}_s \end{Bmatrix}$$

the aeroelastic state equations were rewritten as:

$$\{\dot{\hat{x}}\} = \begin{bmatrix} A_a & B_a H_1 C_s \\ q B_s C_a & A_s + q B_s D_a H_1 C_s \end{bmatrix} \{\hat{x}\} + \begin{bmatrix} B_a H_2 \\ q B_s D_a H_2 \end{bmatrix} \{\eta_c\}. \quad (4)$$

It was desired that the outputs of the aeroelastic state-space system be physical vertical accelerations ($\hat{\delta}$) at 21 locations along the wing (see Fig. 6 for a visualization of these sensor locations). These values were extracted by transforming the modal acceleration vector according to:

$$\{\hat{\delta}\} = [F] \{\hat{\eta}\} \quad (5)$$

where F is a transformation matrix that converts modal accelerations to physical accelerations at the specified 21 points along the wing. Modal accelerations ($\hat{\eta}$) can be defined by isolating certain elements within \hat{x}_s :

$$\{\hat{\eta}_s\} = [H_3] \{\hat{x}_s\} \quad (6)$$

where

$$[H_3] = \begin{bmatrix} 0 & I \end{bmatrix}. \quad (7)$$

Combining Eqs. 5, 6, and 7 yields the final output equations:

$$\{\hat{\delta}\} = \begin{bmatrix} q F H_3 B_s C_a & F H_3 A_s + q F H_3 B_s D_a H_1 C_s \end{bmatrix} \{\hat{x}\} + \begin{bmatrix} q F H_3 B_s D_a H_2 \end{bmatrix} \{\eta_c\}. \quad (8)$$

Equations 4 and 8 constitute the state-space system for ROM-II.

A summary of inputs and outputs for ROM-II may be helpful at this point. Recall that the case presented here used 4 structural modes and 2 control modes to model the structural dynamics. The structural model, therefore, had 4 inputs (generalized aerodynamic forces on each structural mode, scaled by some dynamic pressure) and 4 outputs (modal displacements for each of the structural modes). The aerodynamic ROM used in ROM-II had 6 inputs (control surface and structure modal displacements) and 4 outputs (generalized aerodynamic forces on each structural model). The combined aeroelastic model had 2 inputs (control surface modal displacements) and 21 outputs (accelerations at physical points along the wing).

The second family of ROMs, designated ROM-III, was assembled as shown in Fig. 7. The aerodynamic ROMs were derived from 2 different training simulations, one where the control modes were excited and modal forces were output on the structure and another where the structural modes were excited and modal forces were output on the structure. Having separate training simulations for structural modes and control modes has been shown to model the unforced (free) response better [30]. The aeroelastic state-space model for ROM-III was derived similarly to that of ROM-II, with

slight differences for the fact that there were two aerodynamic ROMs. The final aeroelastic state-space system was found to be:

$$\begin{Bmatrix} \hat{\dot{x}} \\ \hat{\dot{\delta}} \end{Bmatrix} = \begin{bmatrix} A_{as} & 0 & B_{as}C_s \\ 0 & A_{ac} & 0 \\ qB_sC_{as} & qB_sC_{ac} & A_s + qB_sD_{as}C_s \end{bmatrix} \begin{Bmatrix} \hat{x} \\ \hat{\delta} \end{Bmatrix} + \begin{bmatrix} 0 \\ B_{ac} \\ qB_sD_{ac} \end{bmatrix} \begin{Bmatrix} \eta_c \end{Bmatrix} \quad (9)$$

$$\begin{Bmatrix} \hat{\dot{\delta}} \end{Bmatrix} = \begin{bmatrix} qFH_3B_sC_{as} & qFH_3B_sC_{ac} & FH_3A_s + qFH_3B_sD_{as}C_s \end{bmatrix} \begin{Bmatrix} \hat{x} \\ \hat{\delta} \end{Bmatrix} + \begin{bmatrix} qFH_3B_sD_{ac} \end{bmatrix} \begin{Bmatrix} \eta_c \end{Bmatrix} \quad (10)$$

where $A_{as}, B_{as}, C_{as}, D_{as}$ and $A_{ac}, B_{ac}, C_{ac}, D_{ac}$ are the state-space matrices describing the aerodynamic ROM involving the structural and control modes, respectively.

Again, a brief review of inputs and outputs may be helpful for ROM-III. The structural model remained equivalent to that of ROM-II. The aerodynamic ROM associated with structural modes had 4 inputs (structural modal displacements) and 4 outputs (aerodynamic forces acting on the 4 structural modes). The aerodynamic ROM associated with the control modes had 2 inputs (control surface modal displacements) and 4 outputs (aerodynamic forces acting on the 4 structural modes). The combined aeroelastic model had 2 inputs (control surface modal displacements) and 21 outputs (accelerations at physical points along the wing).

D. Phase 3: Observer and Controller Design

In order to implement AFS, a full-state feedback controller was developed. Full-state feedback was chosen due to its simple mathematical properties over other control schemes. Full-state feedback involves multiplying the states (or estimates of the states) by some feedback gain matrix, K , to produce some control input vector, u . To do this using the aerodynamic states from FUN3D was not directly possible as the number of states was extremely large (modal structural data, but also velocity, pressure, and density information at each node), and the requisite matrices were not easily accessible. Instead, a small amount of data was taken from FUN3D at each time step (25 modal accelerations, \ddot{y}_j , converted to 21 vertical acceleration measurements along the wing, δ), and used to estimate the full state vector of a stand-alone linear aeroelastic model. This linear aeroelastic state was then used to produce the commanded control surface rotations, which were fed to FUN3D. The development of these observers and controllers followed the methods presented in Ref. [31], and is essentially a linear quadratic Gaussian (LQG) controller. The stand-alone linear aeroelastic observer was integrated in time alongside FUN3D, and was assumed to be of the form:

$$\dot{\hat{x}} = A\hat{x} + Bu + L(\delta - \hat{\delta}) \quad (11)$$

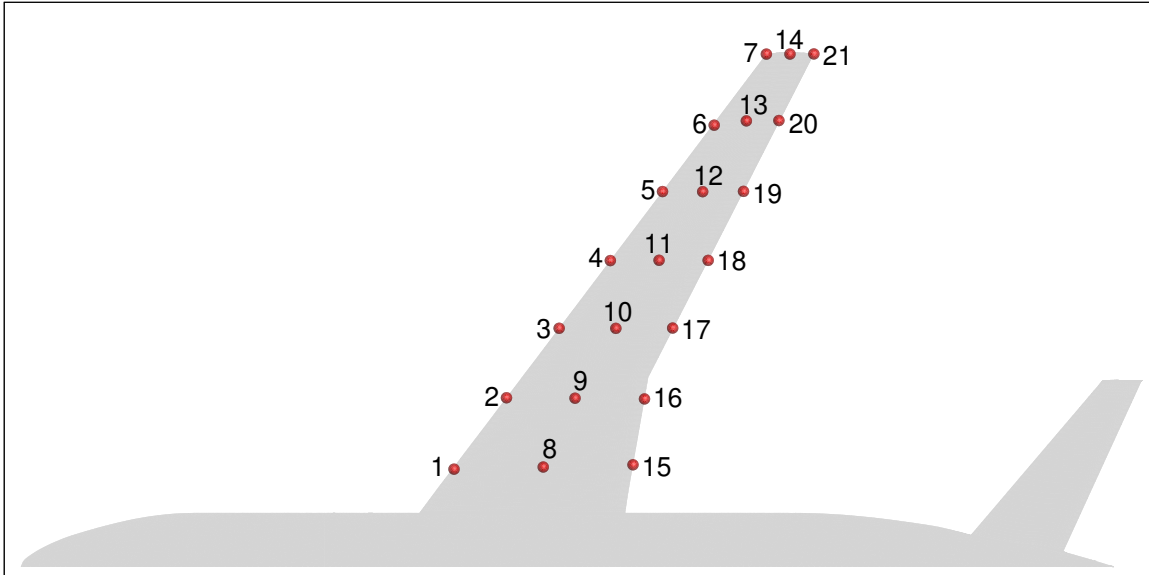


Fig. 6 Sensor locations.

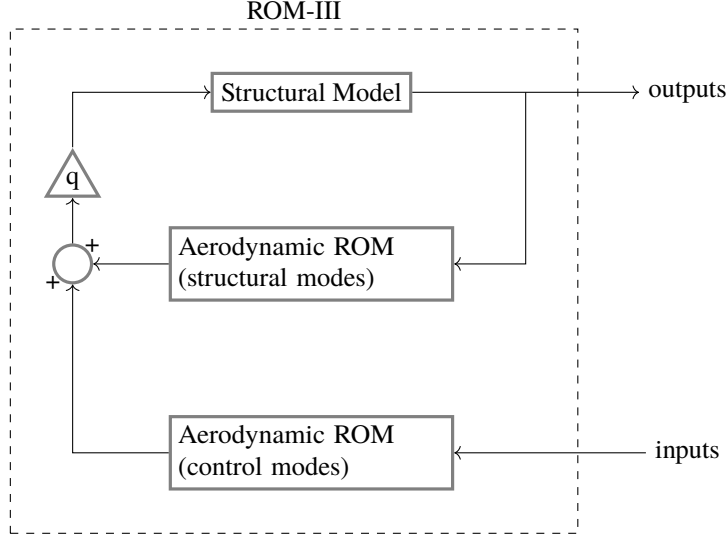


Fig. 7 Block diagram for ROM-III. Two aerodynamic ROMs were produced and used in ROM-III: one strictly for structural modes and the other strictly for control modes. This was done to decouple any unwanted interactions between the control and structural modes in the training simulations. Inputs for ROM-III were two control mode displacements and outputs were modal displacements of four modes of interest.

where u is a 2×1 vector of control surface rotations and \hat{x} is a vector of linear aeroelastic states, generated from the linearized aeroelastic ROM (or another linear aeroelastic model); $\ddot{\delta}$ is the 21×1 vector of vertical accelerations along the wing, which comes from FUN3D, and $\hat{\delta}$ is its estimate generated from the linear aeroelastic model. As previously mentioned, L is an observer gain matrix whose derivation is described below. When L is properly designed, the error between the FUN3D accelerations and the linear accelerations ($\ddot{\delta} - \hat{\delta}$) will quickly decay to zero during the unsteady simulation.

Having obtained a compact aeroelastic state, feedback was assumed of the form:

$$u = -K\hat{x}$$

The feedback gain matrix (i.e., the controller), K , was found such that a performance index, J , was minimized:

$$J = \frac{1}{2} \int_0^{\infty} (\hat{x}^T Q \hat{x} + u^T R u) dt \quad (12)$$

where $\frac{1}{2} \int_0^{\infty} \hat{x}^T Q \hat{x} dt$ represents the energy of some combination of the states, and $\frac{1}{2} \int_0^{\infty} u^T R u dt$ represents the energy necessary for control; Q is a state weighting matrix; R is a control weighting matrix. Larger values of R relative to Q penalize the controls more. The goal was to minimize the energy in the states without using too much energy in the control effort. The L matrix from Equation 11 was generated using a similar equation to the J performance index with R and Q being white noise inputs. A block diagram representation of the final observer and controller was shown in Fig. 1.

When minimizing Eq. 12 to find a controller, K , the choices for the Q and R matrices have a significant influence on the positions of the closed-loop plant poles. Similarly, when minimizing Eq. 12 to find an observer gain matrix, L , the choices for the Q and R matrices influence the closed-loop observer poles. Thus, there were two sets of Q and R matrices associated with K and L , respectively. The method used here to find Q and R follows that found in Ref. [31], which was to assume some performance output — which was to be kept small — of the form $z = H\hat{x}$. To find the Q and R matrices associated with the controller, K , z was chosen to be the modal velocities of the structural modes; therefore, H was simply a matrix that isolated the modal velocity terms from the estimated state vector \hat{x} and zeroed out all other terms. Then, the performance index was redefined as:

$$J = \frac{1}{2} \int_0^{\infty} (z^T \bar{Q} z + u^T R u) dt$$

where $\bar{Q} = H^T \bar{Q} H$; \bar{Q} was a diagonal matrix containing $q_i = \frac{1}{z_{max_i}^2}$ and z_{max_i} was some maximum allowable deviation in each element of z for i equal to 1 to the number of elements in z . In Ref. [31], R is defined in a similar fashion as \bar{Q} , but for control inputs. However, in order to yield satisfactory AFS behavior from the combined system, R was tuned to ensure that the closed-loop observer poles were faster than the closed-loop plant poles. This was done simply by defining $R = \rho I$ and changing ρ until the desired time response was obtained. In this work, the performance of several controllers derived from different linear aeroelastic models was to be compared. In order to show a fair comparison, the values for \bar{Q} and R associated with both K and L were identical across all controllers.

III. Results

With state-space matrices modeling the aeroelastic response of the system, and with an observer gain matrix and a feedback gain matrix, a case demonstrating the ability to suppress flutter was set up by assembling all of the matrices and a computational aeroelastic simulation as shown in Fig. 1. FUN3D was used to execute the computational aeroelastic simulation (i.e., the FUN3D aeroelastic simulation was the plant) by making use of its internal modal structural dynamics solver with the same geometry as was used in the training simulation. The demonstration case involved assigning a nonzero initial value to the first structural modal velocity at an unstable dynamic pressure of 35 kPa. This demonstration case was restarted from a converged static aeroelastic case, following the same process described in Section II.B. Without a control system, such a case would be unstable — modal displacements would grow increasingly large. It was shown, however, that the system could be stabilized using the observer and controller setup described so far in this paper.

A total of 25 structural modes were used, which was more than the 4 modes that were used to generate the ROMs. Using less modes for the ROM gave insight into how well those 4 modes captured the physics of the problem as opposed to using 25 modes. Furthermore, it demonstrated that effective controllers could be designed based on a subset of modes. This means that, when comparing results from the plant (FUN3D) to the observer (ROM), the comparison will be conservative.

Additionally, as was mentioned in Section II.B, no validation or verification data will be presented here for the computational mesh beyond a mesh convergence study shown in Ref. [27]. Since the intent here is not to demonstrate the correctness of the computational solution, but rather to demonstrate the feasibility and potential for this particular controller/observer design process, no additional validation or verification was deemed necessary.

Figure 8 compares the uncontrolled case to cases where controllers and observers were implemented as shown in Fig. 1. Immediately noticeable are the failures of ROM-IIb and ROM-IIc to suppress the instability, while the rest of the observer/controller combinations appear to arrest the instability well. The failures of ROM-IIb and ROM-IIc could be corrected by changing the values of the feedback gains, but this would invalidate a fair comparison across the several controllers, which all used the same gains. It is, however, instructive to visualize the placement of the closed-loop plant poles relative to the closed-loop observer poles and compare them to a case that successfully suppressed the instability. This is done in Fig. 9. Note that, for a successful case (Fig. 9a), the slowest closed-loop observer pole was 2.5-3 times faster than the slowest closed-loop plant pole. In contrast, the slowest closed-loop observer pole in the unsuccessful case (Fig. 9b) was nearly as slow as the slowest closed-loop plant pole. Again, the feedback gains could be changed to improve pole placement, but in order to fairly compare all control efforts, the gains on all other controllers would need to match. (Note that Fig. 9 shows that the combined system for ROM-IIc is stable, but when used as an observer, ROM-IIc is unstable via FUN3D, as shown in Fig. 8.)

Another explanation as to why ROM-IIb and ROM-IIc failed to suppress the instability might be found by examining the root locus plots of each ROM. This was done in Ref. [30], and Fig. 10 is recreated here from that paper. Figure 10a shows a significant sensitivity of the dynamics of the ROM-II systems to the conditions at which their respective training simulations occurred. That sensitivity was not as apparent in ROM-III systems, as shown in Fig. 10b. Hence, this research suggested that having separate aerodynamic ROMs for structural and control may capture free response (where no control input signals are present) physics better. Note that ROM-IIb and ROM-IIc predict mode 1 going unstable, while all the other ROMs predict mode 2 as the unstable mode. This difference may contribute to the inability of controllers derived from ROM-IIb and ROM-IIc to suppress the instability.

Ultimately, the fact that they failed is not disturbing since the closed-loop observer poles were only slightly faster than the closed-loop plant poles, which can be changed by manipulating the controller and observer gains, as mentioned previously. These failures may indicate, however, that aeroservoelastic ROMs that employ a single aerodynamic ROM where structural and control modes are included together may not be as desirable for aeroservoelastic applications as those that employ separate aerodynamic ROMs for structural and control modes.

Figure 11 compares the modal displacements for modes 1, 2, 3, and 8 as predicted by the observer (which are embedded in the linear state vector, \hat{x}) to the modal displacements for the same modes as predicted by FUN3D. Additionally, Fig. 12 shows the same type of comparison for physical vertical accelerations at sensor locations 11 and 14 (see Fig. 6). Again, all the observer predictions appear to match the FUN3D predictions well except for ROM-IIb and ROM-IIId. Figure 13 was generated by taking the fast Fourier transform of the data in Fig. 12 and shows the physical vertical accelerations in the frequency domain at sensor locations 11 and 14. It appears that, with the exception of ROM-IIb and ROM-IIId around 15 Hz, the various observers capture content in the lower frequencies fairly well, whereas they fail at higher frequencies, which is to be expected. Changing the feedback gains could improve performance for ROM-IIb and ROM-IIId since the observer's input is the negative product of the predicted states, \hat{x} , and the feedback gain matrix, K . Additionally, changing the gains of the observer gain matrix, L , may improve the observer's ability to predict modal displacements. However, the same problem arises as when considering an adjustment to feedback gains – doing so invalidates a fair comparison across controllers.

IV. Conclusion

The procedure for developing a ROM-based AFS system suitable for transonic flight speeds was presented in this paper. Using the method described, several AFS systems were developed and their performance was demonstrated. For each phase in the procedure, several important considerations were presented. In the aerodynamic training simulation phase, care must be taken when determining the length of the Walsh function input and the step size of the simulation. These parameters determine the frequency range that is captured in the simulation. The magnitude of the Walsh function

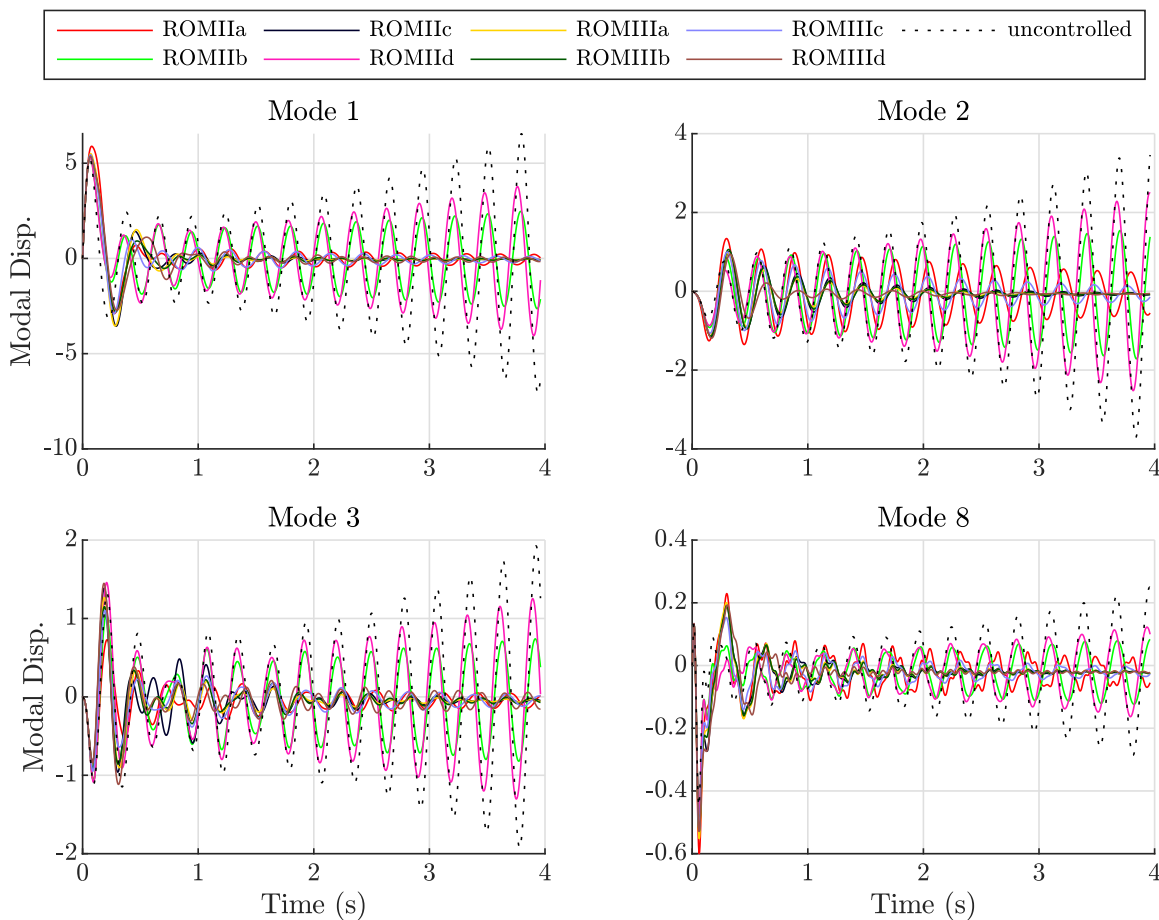


Fig. 8 Comparison of AFS results using the several different aeroservoelastic models described. Most of the models are able to suppress the instability, but ROM-IIb and ROM-IIId are not. Note this case was run at an unstable dynamic pressure of 35 kPa.

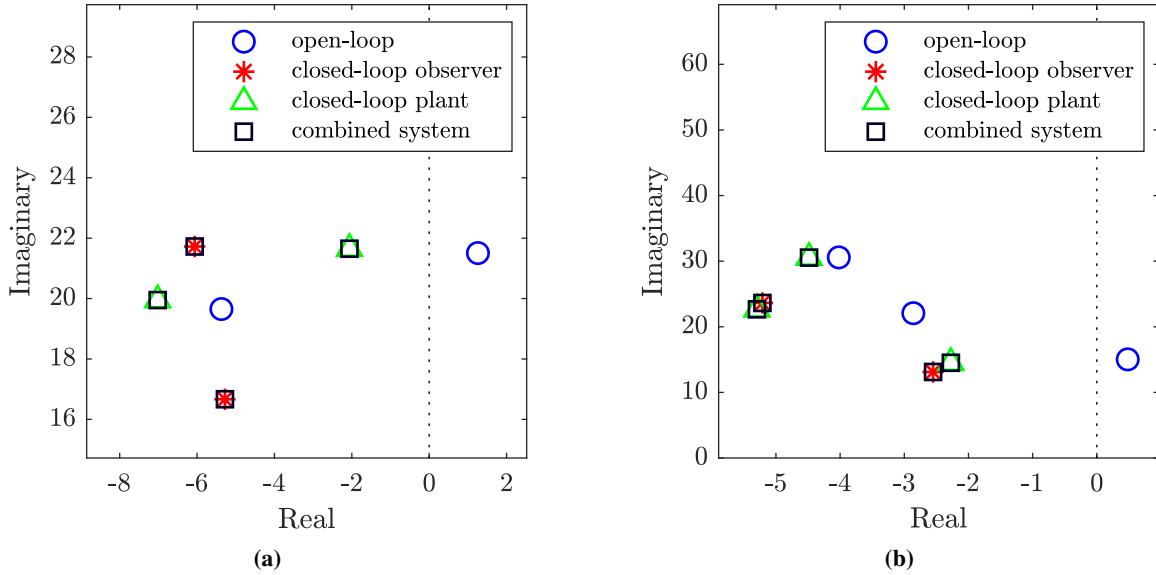


Fig. 9 Visualization of pole placement for (a) ROM-IIa and (b) ROM-IIId.

input is also an important consideration in this phase. In general, these values are configuration-dependent. The conditions at which the training simulation is run also played an important role in the development of the aerodynamic ROMs.

In the ROM generation phase, two methods for producing a unified aeroelastic model were discussed—one with an aerodynamic ROM that includes all structural and control modes, and another where there are separate aerodynamic ROMs for structural and control modes. Previous research suggested that having separate aerodynamic ROMs for structural and control modes captured free response physics better. Although not conclusive, the closed-loop results presented here may help to confirm this, as two AFS systems that employed an aerodynamic ROM where structural and control modes were included together failed to suppress flutter.

In the last phase, observer and controller design, emphasis was placed on the selection of weighting matrices Q and R . A method was presented that determines values for these weighting matrices based on some performance output that the user selects. Ultimately, however, Q and R determine the placement of closed-loop poles for the observer and plant. It was shown that the closed-loop observer poles should be at least 2.5-3 times faster than the closed-loop plant poles to achieve AFS.

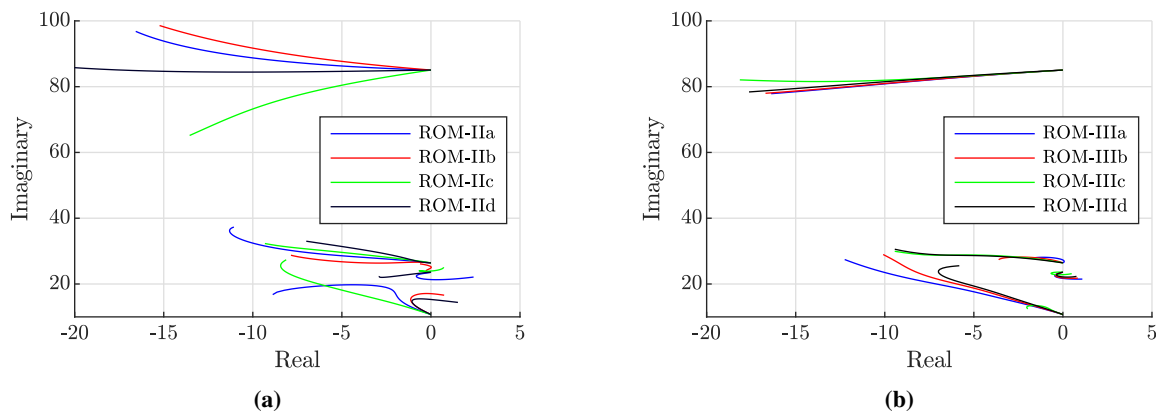


Fig. 10 Root locus plot for (a) ROM-II and (b) ROM-III across several training simulation conditions. Note that these plots do not involve control surface motion.

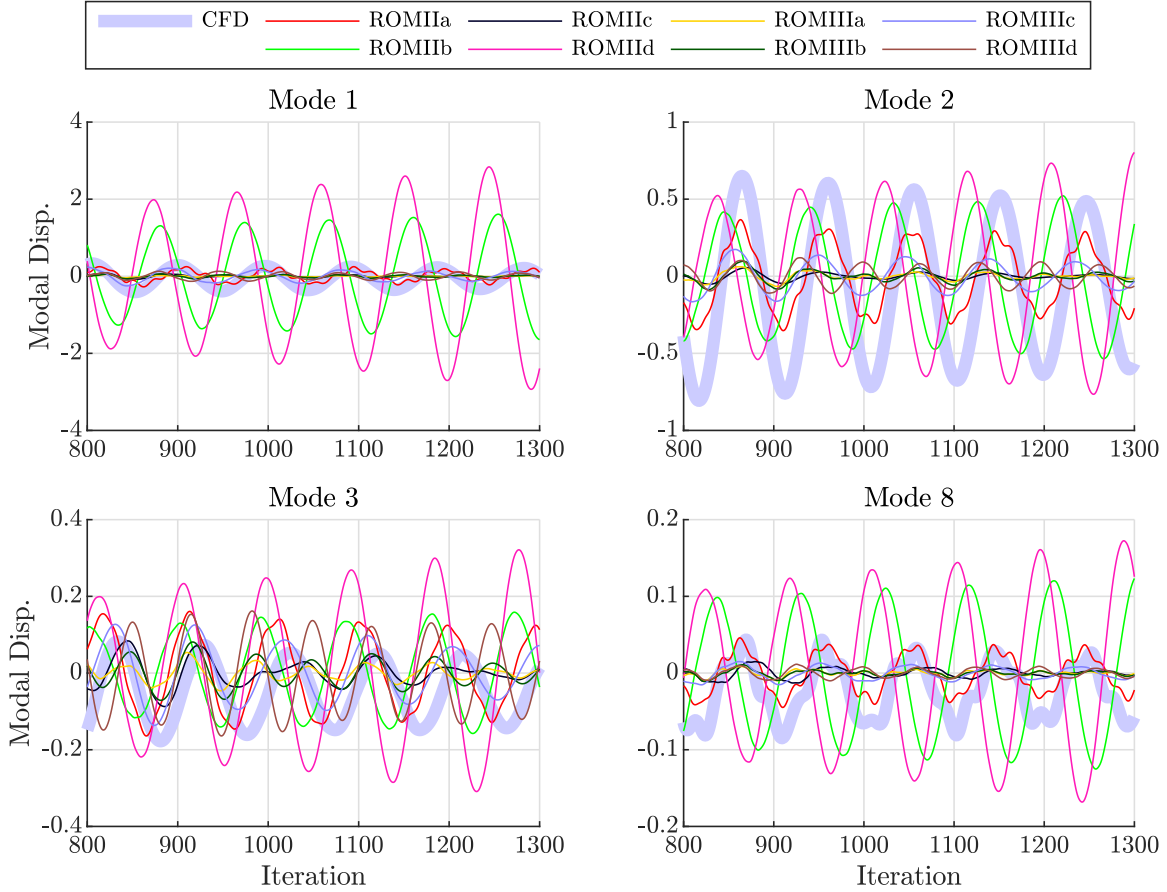


Fig. 11 Comparison of modal displacement predictions from the observer against modal displacements from the plant during a closed-loop simulation.

A total of eight aeroservoelastic AFS systems were generated and compared in this paper. Estimates of the structural states from the observer compared favorably with the same structural parameters as computed by the plant for all but two of these AFS systems. Both of these systems, those associated with ROM-IIb and ROM-IIId, were also unsuccessful in suppressing flutter. The fact that they failed is not disturbing since the closed-loop observer poles were only slightly faster than the closed-loop plant poles, which can be changed by manipulating the controller and observer gains. However, it may indicate (again) that aeroservoelastic ROMs that employ a single aerodynamic ROM where structural and control modes are included together may not be as desirable for aeroservoelastic applications as those that employ separate aerodynamic ROMs for structural and control modes.

The ROMs presented in this paper did not include rigid body modes. In the future, the methods and conclusions presented here will be used to develop aeroservoelastic models that include rigid body modes.

Acknowledgments

This work is funded by the NASA Advanced Air Transport Technologies project. The authors would like to acknowledge the support of Dr. Steven Massey for his help in generating computational meshes used in this work.

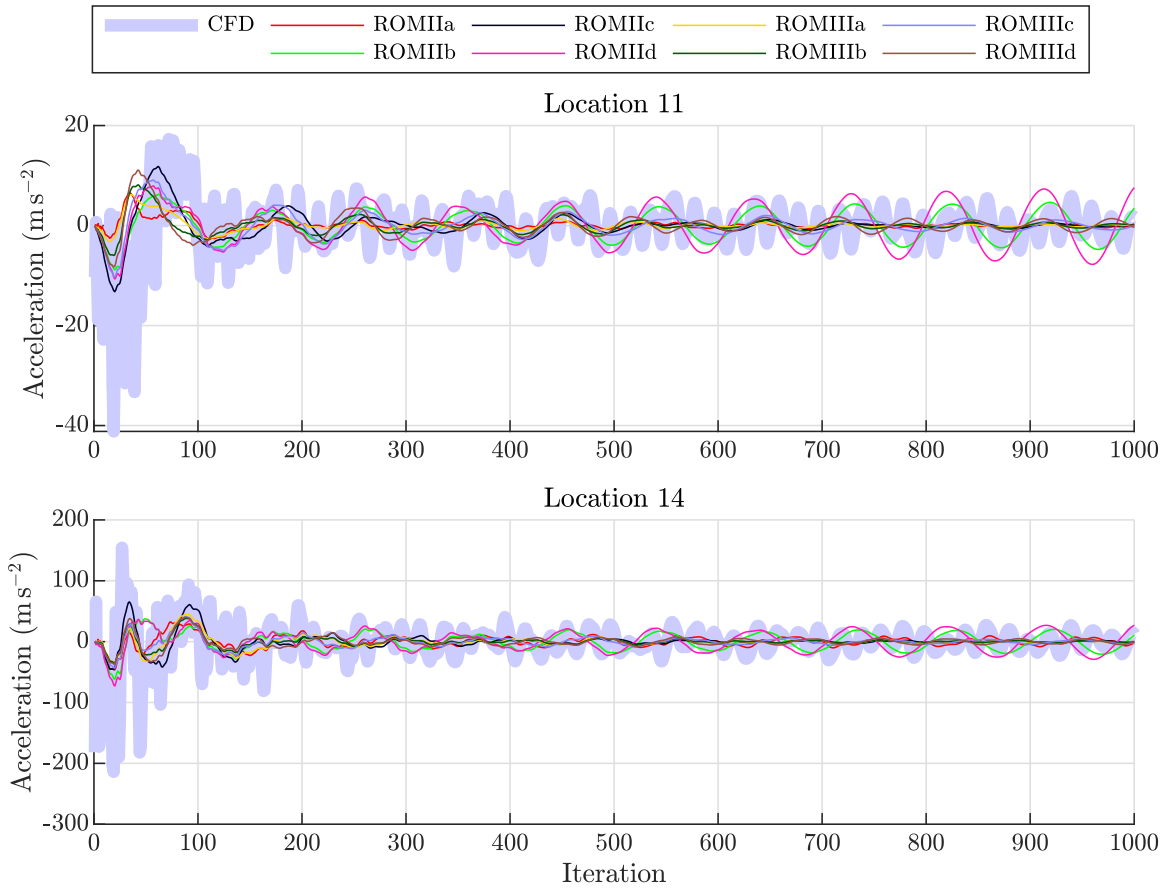


Fig. 12 Comparison of physical vertical acceleration predictions from the observer against physical vertical accelerations from the plant at sensor locations 11 and 14 during a closed-loop simulation.

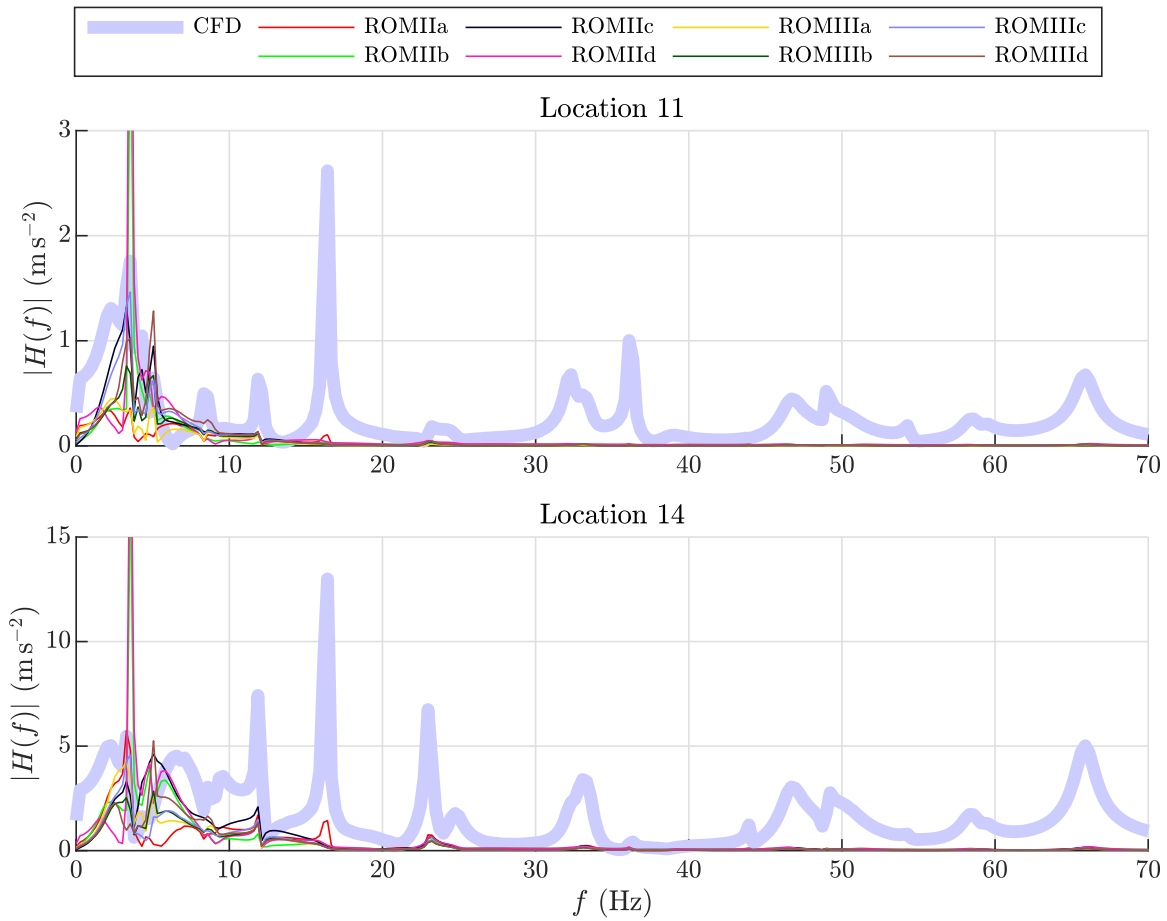


Fig. 13 Frequency domain comparison of physical vertical acceleration predictions from the observer against physical vertical accelerations from the plant at sensor locations 11 and 14 during a closed-loop simulation.

References

- [1] Livne, E., "Aircraft Active Flutter Suppression: State of the Art and Technology Maturation Needs," *Journal of Aircraft*, 2017, pp. 1–41.
- [2] Waite, J., Stanford, B., Bartels, R. E., Silva, W. A., and Massey, S. J., "Active Flutter Suppression Controllers Derived from Linear and Nonlinear Aerodynamics: Application to a Transport Aircraft Model," *2018 Applied Aerodynamics Conference*, Atlanta, GA, 25-29 Jun 2018.
- [3] Mukhopadhyay, V., "Flutter Suppression Control Law Design and Testing for the Active Flexible Wing," *Journal of Aircraft*, Vol. 32, No. 1, 1995, pp. 45–51.
- [4] Lu, Z., Cui, Y., Schneider, D., Zhao, Z., Chen, X., Lai, K. L., and Lum, K.-Y., "Aeroelastic Responses Identification of a High-Aspect-Ratio Flexible Wing Model and Its Active Flutter Suppression Strategy," *57th AIAA/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference*, San Diego, CA, 4-8 Jan 2016.
- [5] Qian, W., Huang, R., Hu, H., and Yonghui, Z., "Active Flutter Suppression of a Multiple-Actuated-Wing Wind Tunnel Model," *Chinese Journal of Aeronautics*, Vol. 27, No. 6, 2014, pp. 1451–1460.
- [6] Block, J. J., and Strganac, T. W., "Applied Active Control for a Nonlinear Aeroelastic Structure," *Journal of Guidance, Control, and Dynamics*, Vol. 21, No. 6, 1998, pp. 838–845.
- [7] Burnett, E., Beranek, J., Holm-Hansen, B., Atkinson, C., and Flick, P., "Design And Flight Test of Active Flutter Suppression on the X-56A Multi-Utility Technology Test-Bed Aircraft," *The Aeronautical Journal*, Vol. 120, No. 1228, 2016, pp. 893–909.
- [8] Ting, E., Chaparro, D., and Nguyen, N. T., "Development of an Integrated Nonlinear Aeroservoelastic Flight Dynamic Model of the Truss-Braced Wing Aircraft," *58th AIAA/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference*, Grapevine, TX, 9-13 Jan 2017.
- [9] Wang, Y., Wynn, A., and Palacios, R., "Nonlinear Modal Aeroservoelastic Analysis Framework for Flexible Aircraft," *AIAA Journal*, 2016, pp. 3075–3090.
- [10] Zhao, Y., "Flutter Suppression of a High Aspect-Ratio Wing with Multiple Control Surfaces," *Journal of Sound and Vibration*, Vol. 324, No. 3-5, 2009, pp. 490–513.
- [11] Afonso, F., Vale, J., Oliveira, É., Lau, F., and Suleman, A., "A Review on Non-Linear Aeroelasticity of High Aspect-Ratio Wings," *Progress in Aerospace Sciences*, Vol. 89, 2017, pp. 40–57.
- [12] Hallissy, B., and Cesnik, C., "High-Fidelity Aeroelastic Analysis of Very Flexible Aircraft," *52nd AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics and Materials Conference 19th AIAA/ASME/AHS Adaptive Structures Conference 13t*, Denver, CO, 4-7 Apr 2011.
- [13] Brooks, T. R., Kenway, G. K., and Martins, J., "Undeformed Common Research Model (uCRM): An Aerostructural Model for the Study of High Aspect Ratio Transport Aircraft Wings," *35th AIAA Applied Aerodynamics Conference*, Denver, CO, 5-9 Jun 2017.
- [14] Tantaroudas, N. D., and Da Ronch, A., "Nonlinear Reduced-order Aeroservoelastic Analysis of Very Flexible Aircraft," *Advanced UAV Aerodynamics, Flight Stability and Control: Novel Concepts, Theory and Applications*, 2017, p. 143.
- [15] Danowsky, B. P., Thompson, P. M., Farhat, C., Lieu, T., Harris, C., and Lechniak, J., "Incorporation of Feedback Control into a High-Fidelity Aeroservoelastic Fighter Aircraft Model," *Journal of Aircraft*, Vol. 47, No. 4, 2010, pp. 1274–1282.
- [16] Danowsky, B. P., Lieu, T., and Coderre-Chabot, A., "Control Oriented Aeroservoelastic Modeling of a Small Flexible Aircraft using Computational Fluid Dynamics and Computational Structural Dynamics-Invited," *AIAA Atmospheric Flight Mechanics Conference*, San Diego, CA, 4-8 Jan 2016.
- [17] Selitrennik, E., Karpel, M., and Levy, Y., "Computational Aeroelastic Simulation of Rapidly Morphing Air Vehicles," *Journal of Aircraft*, Vol. 49, No. 6, 2012, pp. 1675–1686.
- [18] Lucia, D. J., Beran, P. S., and Silva, W. A., "Reduced-Order Modeling: New Approaches for Computational Physics," *Progress in Aerospace Sciences*, Vol. 40, No. 1-2, 2004, pp. 51–117.
- [19] Zhang, W., Wang, B., Ye, Z., and Quan, J., "Efficient Method for Limit Cycle Flutter Analysis Based on Nonlinear Aerodynamic Reduced-Order Models," *AIAA journal*, Vol. 50, No. 5, 2012, pp. 1019–1028.

- [20] Kou, J., and Zhang, W., “Reduced-Order Modeling for Nonlinear Aeroelasticity with Varying Mach Numbers,” *Journal of Aerospace Engineering*, Vol. 31, No. 6, 2018, p. 04018105.
- [21] Zhang, W., Kou, J., and Wang, Z., “Nonlinear Aerodynamic Reduced-Order Model for Limit-Cycle Oscillation and Flutter,” *AIAA Journal*, 2016, pp. 3304–3311.
- [22] Silva, W. A., “AEROM: NASA’s Unsteady Aerodynamic and Aeroelastic Reduced-Order Modeling Software,” *Aerospace*, Vol. 5, No. 2, 2018.
- [23] Silva, W. A., and Bartels, R. E., “Development of Reduced-Order Models for Aeroelastic Analysis and Flutter Prediction Using The CFL3Dv6.0 Code,” *Journal of Fluids and Structures*, Vol. 19, No. 6, 2004, pp. 729–745.
- [24] Lieu, T., Farhat, C., and Lesoinne, M., “Reduced-Order Fluid/Structure Modeling of a Complete Aircraft Configuration,” *Computer Methods in Applied Mechanics and Engineering*, Vol. 195, No. 41-43, 2006, pp. 5730–5742.
- [25] Kenway, G., Kennedy, G., and Martins, J., “Aerostructural Optimization of the Common Research Model Configuration,” *15th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference*, Atlanta, GA, 16-20 Jun 2014.
- [26] Biedron, R., Carlson, J.-R., Derlaga, J., Gnoffo, P., Hammond, D., Jones, W., Kleb, B., Lee-Rausch, E., Nielsen, E., Park, M., Rumsey, C., Thomas, J., and Wood, W., “FUN3D Manual: 13.4,” Tech. Rep. TM-2018-220096, NASA, 2018.
- [27] Stanford, B., and Massey, S. J., “Uncertainty Quantification of the FUN3D-Predicted Flutter Boundary on the NASA CRM,” *58th AIAA/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference*, Grapevine, TX, 9-13 Jan 2017.
- [28] Pacheco, R., and Steffen Jr, V., “Using Orthogonal Functions for Identification and Sensitivity Analysis of Mechanical Systems,” *Modal Analysis*, Vol. 8, No. 7, 2002, pp. 993–1021.
- [29] Juang, J.-N., and Pappa, R. S., “An Eigensystem Realization Algorithm for Modal Parameter Identification and Model Reduction,” *Journal of Guidance, Control, and Dynamics*, Vol. 8, No. 5, 1985, pp. 620–627.
- [30] Waite, J., Stanford, B., Bartels, R. E., and Silva, W. A., “Reduced Order Modeling for Transonic Aeroservoelastic Control Law Development,” *AIAA Scitech 2019 Forum*, 2019, p. 1022.
- [31] Stevens, B. L., Lewis, F. L., and Johnson, E. N., *Aircraft Control and Simulation: Dynamics, Controls Design, and Autonomous Systems, 3rd Edition*, John Wiley and Sons, 2015.